bus network are supported by an AV/C architecture. The AV/C architecture is used by devices to create, process and/or transmit AV/C command and response data packets.

The target device is controllable by a controller device that initiates desired data transactions. The desired data transactions are IEEE 1394-2000 write transactions, wherein a controller device requests a target device to perform a task. The data transactions are contained within command and response frames of the command and request data packets which are formatted according to the Function Control Protocol (FCP) and then transferred asynchronously between device nodes on the IEEE 1394-2000 serial bus.

A format of a block write packet 30 according to the IEEE 1394-2000 standard is illustrated in Figure 2. The asynchronous block write packet includes a header 31 and a data payload 32. The header 31 includes the fields destination_ID, tl, rt, tcode, pri, source_ID, destination_offset, data_length, extended_tcode and header_crc. The destination_ID field is a sixteen bit field which specifies the node ID of the receiving node to which the packet is addressed. The transaction label field tl is a six bit field that specifies a unique tag for each outstanding transaction from a node. The retry code field rt is a two bit field which specifies whether the packet is a retry attempt and the retry protocol to be followed by the destination node. The transaction code field tcode is a four bit field that specifies the packet format and the type of transaction that is to be performed. For a write request for data block operation the transaction code field value is equal to 0001.

The priority field pri is a four bit field that is used by the back plane. The source-ID field is a sixteen bit field that specifies the node ID of the transmitting node of the packet. The destination offset field is a forty-eight bit field that specifies the forty-eight bits of the destination node address of the request packet. The data length field is a sixteen bit field that specifies the length of the data field of data block payload packets. The extended transaction code field extended_tcode is a sixteen bit field that conventionally is only meaningful if the transaction code field indicates a lock request or lock response packet. The header_CRC field is a thirty-two bit field that is used to perform a cyclical redundancy check (CRC) on the data within the header.

The data portion of the packet includes a data block payload field and a data_crc field. The data_crc field is a thirty-two bit field that is used to perform a cyclical redundancy check (CRC) on the data within the data portion of the packet.

AV/C command and response data packets are transmitted between networked devices in data streams that are made up of one or more discrete data packets having the format illustrated in Figure 2 and described above. The data packets are transmitted over the serial bus and received by a device with the appropriate destination address. Using a read transaction, data at a particular address within a responding node is transferred back to a requesting node. Using a write transaction, data is transferred from a requesting node to a particular address within one or more responding nodes. Using lock transactions, data is transferred from a requesting node to a responding node, processed with data at a particular address within the responding node and the result is then transferred back to the requesting node.

Again referring to Figure 2, the data payload frame 33 is organized into a data sequence of data fields according to the Function Control Protocol (FCP) defined by the standard IEC 61883, Digital Interface For Consumer Audio/Video. The Function Control Protocol frame provides a simple format to encapsulate device command and response data sets within the IEEE 1394-2000 serial bus for asynchronous block read and write data transactions. The payload of the FCP frame 33 is limited to a maximum of 512 bytes.

Figures 3A and 3B show a detailed command FCP data frame 40 and a response FCP data frame 50 formatted in accordance with the standard AV/C protocol. The first data fields in both of the data frames 40 and 50 are the cts data fields 41 and 51, respectively. The cts data fields 41 and 51 hold 4 bits of data each and define the transaction format that is to be used in the FCP frames 40 and 50; the code for the standard AV/C format shown in Figures 3A and 3B is 0000. The ctype data field 42 and the response data field 52 are also 4 bits in length and encode the data packets 40 and 50 for the type of command or response data transaction. For example, a command frame such as the one shown in Figure 3A, may be encoded for a control command, an inquiry command or any other data transaction that is

required. The subunit type and the subunit ID data fields 43 and 53 encode the data packets 40 and 50 for the resource subunit within the device that is being used to execute the command data set. For example, the command packet may be issued to start a display of a video monitor, to turn on/off a tuner, turn on/off a recorder and the like. Since several subunit resources may belong to the same device or belong to the same device node address, the subunit type and ID is used to distinguish them. The opcode data fields 44 and 54 code the data packets 40 and 50 for the device operation to be executed and the operand data fields define the parameters of the operation to be executed.

As described above, the ctype data field 42 is 4 bits in length and encodes the command data packet 40 for the type of command data transaction included within the data packet 40. Table I below includes the different types of commands specified and the corresponding value for each command.

VALUE	COMMAND TYPE
0	CONTROL
1	STATUS
2	SPECIFIC INQUIRY
3	NOTIFY
4	GENERAL INQUIRY
5-7	Reserved for future specification
$8\text{-}F_{16}$	Reserved for response codes

Table I: AV/C Command Types

As described above, the response data field 52 is 4 bits in length and encodes the response data packet 50 for the type of response data transaction included within the response packet 50. Table II below includes the different types of responses specified and the corresponding value for each response.